Installable LPC Debug Module Design Guide

Revision History

Revision	Date	History
0.5	11/5/99	First Internal Review Draft
0.6	11/11/99	• The Debug module mechanical attachment changed to flat cable. • Serial EEPROM added to module for BIOS discovery and programming support.
0.65	11/15/99	• Added SPD addresses A1, A0 to connector. • New connector pin out. • Added connector layout and dimensions.
0.70	11/23/99	• Added EEPROM data structure specification. Dropped IPMI FRU compatibility requirement. • Dropped Mobile Requirements. • Added figure for motherboard keep-out. • Added cable connector information.
0.90	12/2/99	• Added table of contents • Swapped byte locations of some EEPROM data locations for easier programming interface. • Added requirement for jumper to module EEPROM A2 input for in-circuit programming support. • Added BIOS Flow chart
1.0	12/17/99	• Added Table of Figures and Table of Tables

TABLE OF FIGURES

TABLE OF TABLES

Purpose

This guide describes how to design a debug port for a PC that has no PC-AT* serial COM port. The PC-AT serial COM port has been used by low level debuggers, such as Operating System KERNAL debuggers, as the connection point between the PC under test and the debugger console. The PC-AT serial COM port is no longer a feature on legacy free and legacy reduced PC's.

Requirements

The following requirements were used to arrive at this specification for the debug module.

- Minimize the Bill of Material (BOM) Cost for the PC platform with a debug port.
- The debug port is available on all production hardware.
- Uses Standard interfaces to connect the debug console to the PC under test.
- No silicon design required for quick industry enabling.
- Does not limit the user h/w configurations of the PC system under test.
- Minimize the processor and memory overhead of the debug data stream on the PC under test.
- Be a private resource for the Operating System.
- Easily discovered and enumerated by the Operating System.
- Support one full duplex, 57,600 bits per second serial data pipe, at a minimum.

The debug module does not meet the following debug port requirement:

- Be available on an external connector

System Overview

A system diagram of the debug module is shown in Figure 1 below.

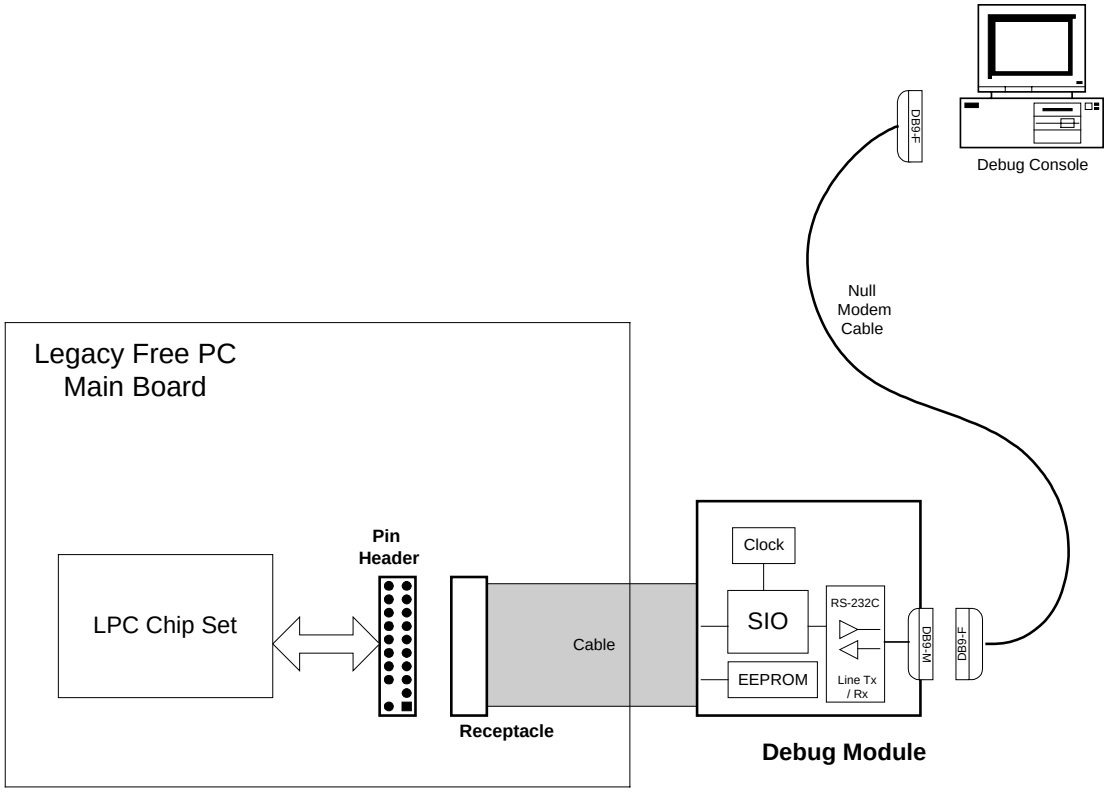

Figure 1: System Diagram

The module consists of a serial communications port, implemented with a standard 16550 UART register interface. The serial communications port registers are not allowed to appear at a legacy COM port I/O addresses and must be reported to the Operating System using a new Advanced Configuration and Power Interface (ACPI) table.

To keep the motherboard cost of the debug module to a minimum, and to minimize impact to existing chassis designs, the debug module connects to the system using a cable.

To minimize the impact to the motherboard, the module interface is placed on the Intel® Low Pin Count (LPC) interface. No LPC 16550 UART is available commercially. To implement the module, an LPC Super I/O device (SIO) must be used. The LPC SIO used must have the following attributes:

- Its registers must be Plug and Play Compatible.
- All legacy controllers (including the 8042) and interfaces in the SIO must be hardware disabled following a PCI reset.

An I2C serial EEPROM is required on the debug module to provide the BIOS with the required information to configure the COM port in the SIO correctly. The SIO programming information and method is specified in the BIOS requirements section of this document. Using the serial EEPROM to specify the programming method allows any SIO that meets the bulleted requirements above to be used on the debug module. The Serial EEPROM can be assigned the I2C addresses: 1010111xb – 1010100xb by the motherboard. Since only 8, I2C SERIAL EEPROM devices can occupy one SMBUS segment, system

designers must insure there is no conflict between the I2C address assigned to the debug module and other Serial EPROM devices in their system.

The serial EEPROM device must be capable of being written for field upgrade support. A jumper on the module, for write enable control, is an acceptable way to implement this requirement.

Address signal A2 to the EEPROM is pulled-up High on the module for normal operation (electrical level '1'). The debug module should allow this pin to be strapped to electrical '0' for programming the EEPROM in situ, using commercially available I2C bus programmers.

The debug module is defined in such a way that it supports two operating environments:

1) Operation with a legacy free OS and
2) Operation with a legacy OS.

Operation with a legacy-free OS is the intended mode of operation of the Debug module.

Operation with a legacy OS may be required to support legacy-free early design validation and manufacturing test flows. Two connector sizes are therefore specified for the module interface. The smaller connector only supports the debug port function. The larger connector supports the signals needed to have full 8042 controller support: RC#, A20GATE, and SERIRQ.

Legacy operation must only be enabled for operation with a legacy OS. This means a BIOS setup option needs to be supported which turns legacy mode on and off. In legacy operation mode, a PS/2* keyboard and mouse would need to be attached to the debug module, since a legacy-free BIOS is not required to provide USB legacy keyboard emulation. In addition, when in operating the module in legacy mode, the COM port should be programmed by the BIOS to operate as COM1 at I/O address x03F8h.

A null modem cable is required to connect the debug module to the serial port on another PC. The debug module uses a DB9-Male connector that is wired in the standard way for a PC serial COM port.

The remainder of this paper *will specify the following features of the debug Module:*

- Module to Motherboard Interface
- Module Mechanical Interface
- BIOS Requirements
- Module RS-232C Interface

Motherboard Interface

The LPC Module motherboard interface is shown in Figure 2 below.

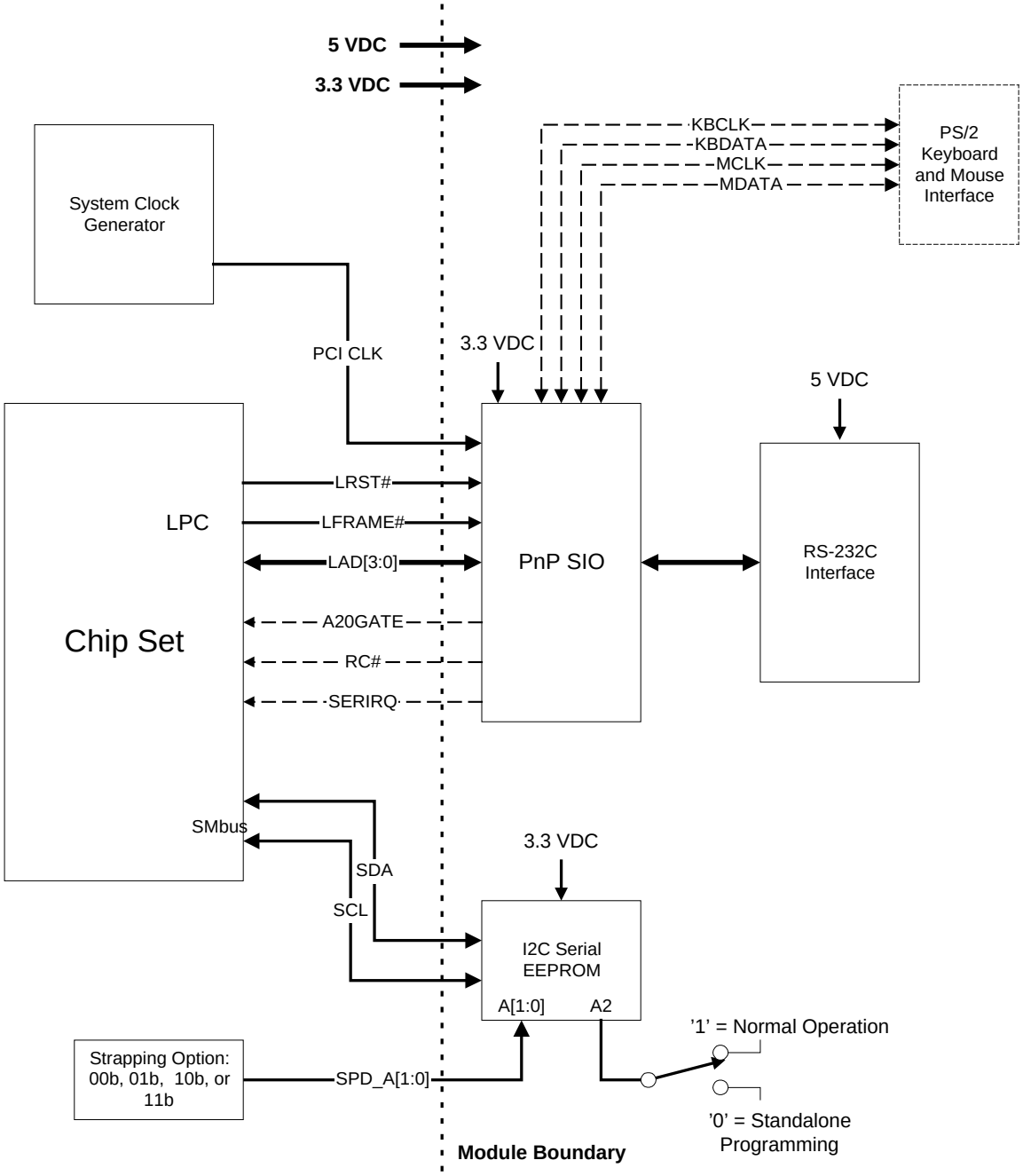

Figure 2: LPC Module Interface Signals

The connector has two lengths: Debug Port (16 pins) and Legacy Extension 20 pins). The 16 Pin connector is intended for systems that will run a legacy-free operating system. The Legacy extension connector is intended for use with both legacy and legacy-free operating systems**.**

The 16 Pin connector consists of the minimum LPC bus signals, an I2C bus, a mechanical key location, and power and ground pins.

The 20 Pin connector adds the 8042 controller legacy signals (RC# and A20GATE) and a Serial Interrupt line, which is used to route IRQ1 and IRQ12 to the motherboard.

The16 pin cable connector and 20 pin header connector are mechanically incompatible. A 16 pin cable connector will not mate with a 20 pin header due to interference between the cable connector body and the additional pins on the 20 pin header. A 20 pin cable connector may or may not be useable with a 16 pin header. This will depend on the mechanical clearance between the motherboard components and the cable connector body.

5 VDC and 3.3 VDC power are required. 3.3 VDC is required for the LPC Interface and for the serial EEPROM. 5 VDC is required for the RS-232C drivers and receivers on the module. The ***power pins must be de-coupled with capacitors both on the module and on the motherboard, at their respective connectors pins,*** to provide an AC signal return path for the signals in the connecting cable. In addition, bulk de-coupling capacitors must be provided on the power planes of the module to compensate for the inductance of the cable connection.

The motherboard connector is a non-shrouded pin header. The motherboard connector uses a missing pin at the position labeled KEYWAY to guarantee proper module cable alignment. The receptacle connector on the module cable connector must have the KEYWAY location plugged to guarantee correct installation.

Signal Descriptions

LPC Interface Signals

Only a brief description of the LPC Interface signals is given here. Please refer to the LPC Interface specification at http://developer.intel.com/design/chipsets/industry/lpc.htm for a full description of these signals

All signals and voltages on the debug module interface are only valid during the ACPI G0/S0, Legacy, G1/S1 and G1/S2 states.

LCLK: This is a 3.3 VDC PCI clock. This clock must meet PCI signal requirements including maximum skew with respect to the other PCI clocks in the system.

LFRAME#: This signal is used to indicate the start and termination of cycles on the LPC interface. This is a 3.3 VDC signal.

LRST#: This is a 3.3 VDC PCI reset.

LAD[3:0]: These are 3.3 VDC Multiplexed Command, Address, and Data.

A20GATE: This signal is an input to the A20GATE input on the motherboard chip set. It is logically combined with the “fast A20” signal to create the A20M# signal to the processor. The motherboard should provide a 10 K ohm pull up resistor on this signal at the chip set input for operation when the module is not present. This signal is only required on the legacy extension connector for legacy 8042 support.

RC#: This is the legacy keyboard reset. It is an input to the chip set where it is logically combined with a register generated INIT signal to create the processor INIT# signal. The motherboard should provide a 10 K ohm pull up resistor on this signal at the chip set input for operation when the module is not present. This signal is only required on the legacy extension connector for legacy 8042 support.

SERIRQ: This is the serialized IRQ signal. It is an output from the module and an input to the motherboard chip set. This signal is required to supply legacy IRQs from the SIO on the debug module to the motherboard chip set. This signal should be terminated with a 10 K ohm pull up resistor on the motherboard for the case where no debug module is installed. This signal is only required on the legacy extension connector.

VCC5: This is a 5 volt DC power rail supplied by the motherboard to the module. The maximum power for this interface is 250 ma.

VCC3: This is a 3.3 volt DC power rail supplied by the motherboard to the module. The maximum power for this interface is 250 ma.

SDA: This is a 3.3 volt I2C data signal.

SCL: This is a 3.3 volt I2C clock signal.

SPDA1: This is a 3.3 volt signal that connects to hardware address input A1 on the serial EEPROM device on the module. The base board must strap this signal High or Low to a value that does not conflict with any other serial EEPROM on the I2C Bus routed to the module.

SPDA0: This is a 3.3 volt signal that connects to hardware address input A0 on the serial EEPROM device on the module. The base board must strap this signal High or Low to a value that does not conflict with any other serial EEPROM on the I2C Bus routed to the module.

Mechanical Requirements

Use of the debug module requires a system chassis that can be operated when open. The motherboard module cable connector (pin header) must be located on the motherboard so that it can accept the module cable connector and cable with out interfering with other baseboard components.

The maximum allowed length of the debug module to motherboard cable is 8 inches.

A debug module will have two external interfaces: A cable for connection to the system under test, and an RS-232C DB9-M, wired as a computer terminal. The Legacy debug Module, in addition to the above external connectors, will have PS/2 style Keyboard and Mouse connectors.

The debug module circuitry must be encased in a non-conductive housing to prevent inadvertent shorting with the system under test.

The module is not capable of hot-plug insertion. All power to the system must be off when installing and removing the module. Industry accepted ESD handling guidelines must be observed when plugging the module into the motherboard to prevent damage to motherboard circuitry.

Module to Motherboard Mechanical Interface

The only mechanical connection between the debug module and the motherboard is the connector. No mechanical guides or retention hardware are required by this specification.

Motherboard Connector

The motherboard connector for desktop systems is a vertical 0.1" x 0.1" pin header with 16 or 20 pins. Pin 4 is voided to insure proper alignment with the mating cable. The header pins are 0.025 inch square posts or a round post with equivalent dimensioning. The header parameters are specified in Table 1. Due to the wide availability of this type of connector, no manufacturer part number is specified.

Table 1: Pin Header Parameters

Connector	Voided Pin Position	Header Parameter "A" (inches)	Number of Circuits
Debug	4	0.70	16
Legacy Extension	4	0.90	20

The connector physical dimensions are specified in Figure 3 below.

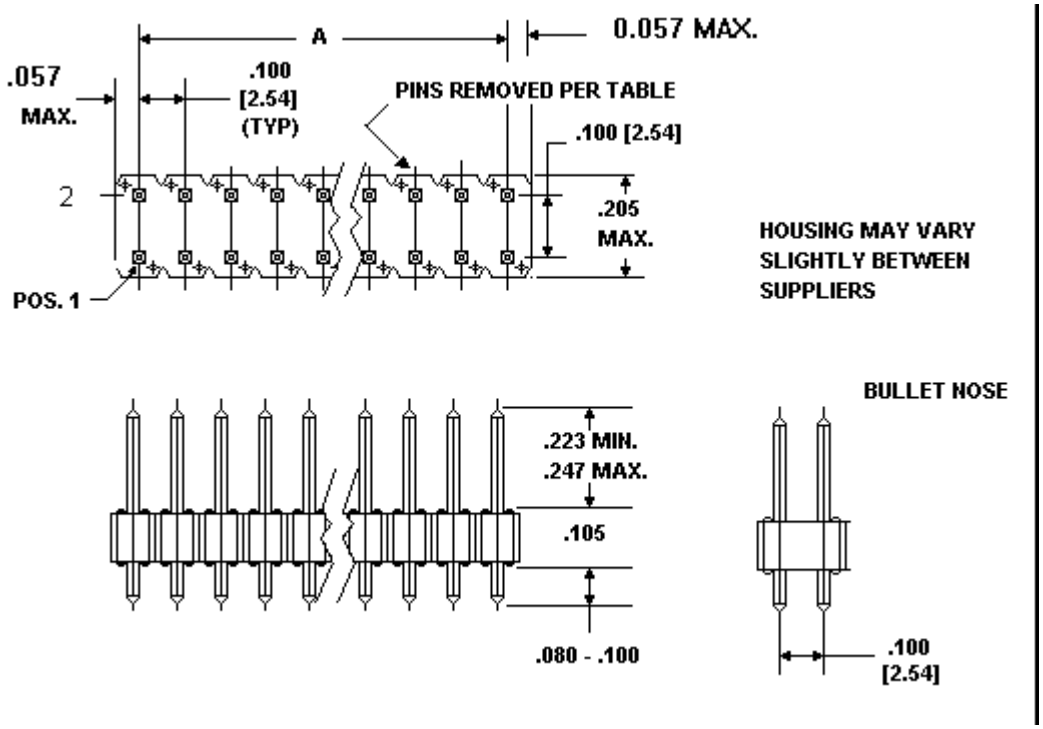

Figure 3: Connector Dimensions

The connector hole pattern is specified in Figure 4 below.

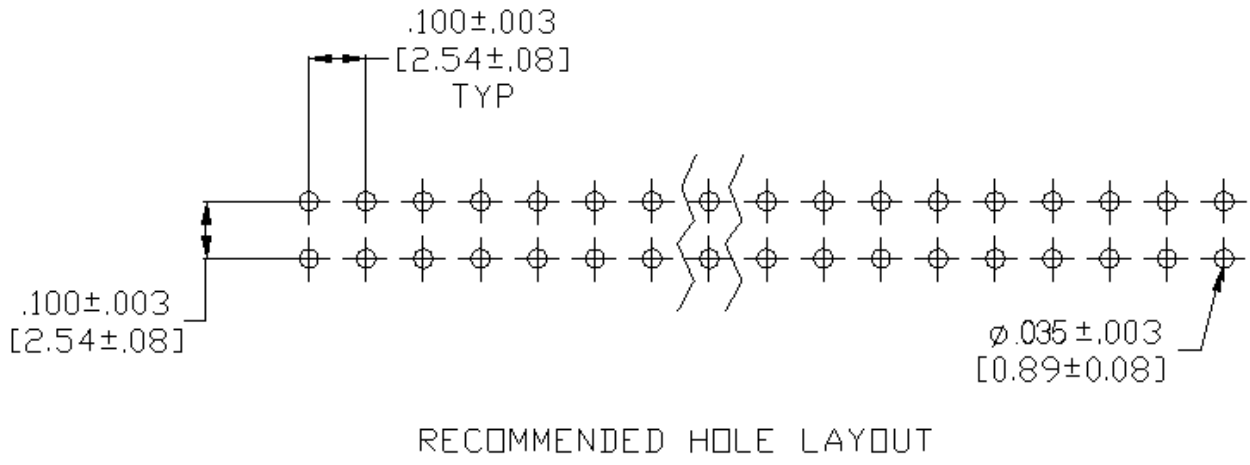

Figure 4: Connector Hole Layout

A physical keep-out around the connector must be observed. The keep out zone is 0.160 inches from the end of the connector hole pattern and 0.080 inches from the side of the connector, measured from the center of any pin. The keep out is required to allow the cable connector to be attached without interference with adjacent components. The keep out is specified in Figure 5.

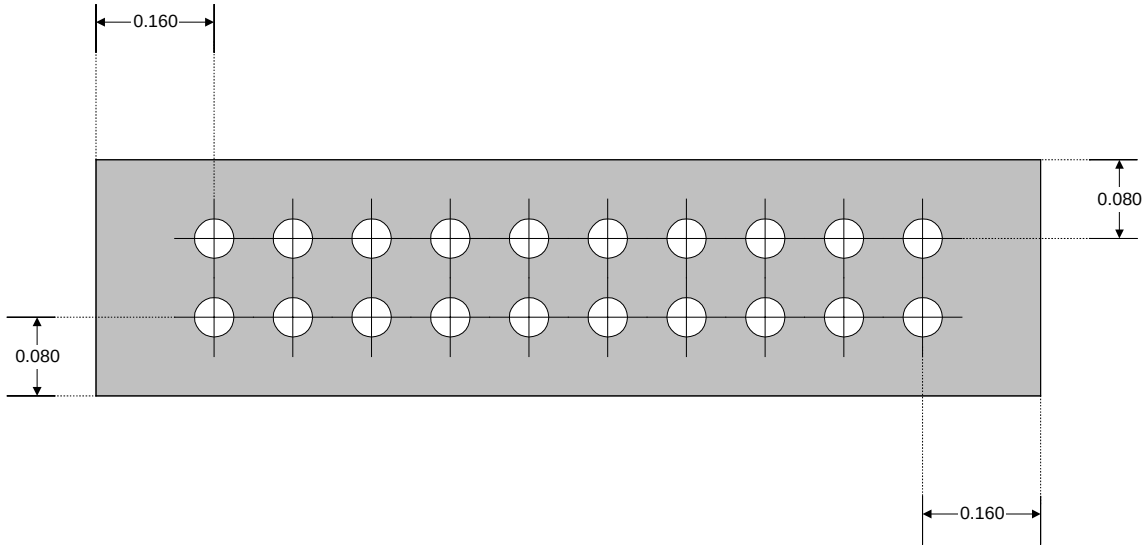

Figure 5: Connector Keep out

The connector numbering scheme is specified in Figure 6 below.

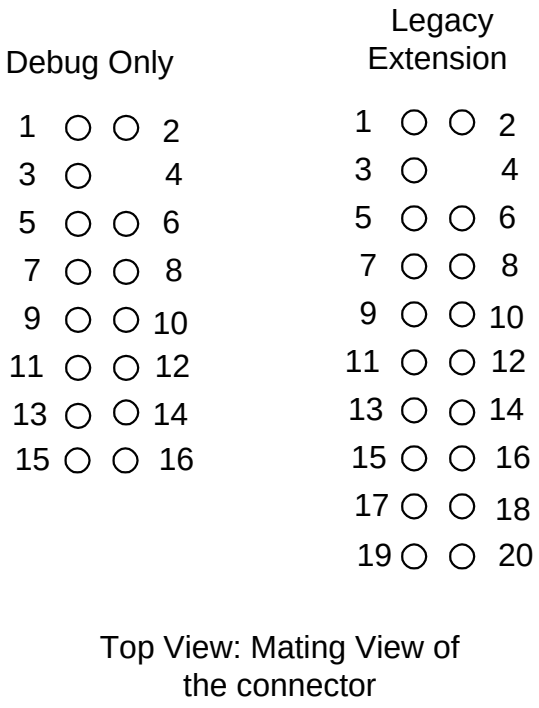

Figure 6: Motherboard Connector Pin numbering

Motherboard Connector Signal Assignment - Desktop

The motherboard connector pin assignments for the debug connector and the legacy extension is specified in Table 2 below.

Table 2: Motherboard Connector Signal Assignment - Desktop

Pin	**Signal**	***Pin***	**Signal**	**Notes**
1	LCLK	2	VSS	Debug Connector
3	LFRAME#	4	KEYWAY	
5	LRST#	6	VCC5	
7	LAD3#	8	LAD2#	
9	VCC3	10	LAD1#	
11	LAD0#	12	VSS	
13	SCL	14	SDA	
15	SPDA1	16	SPDA0	
17	VSS	18	SERIRQ	Legacy Extension
19	RC#	20	A20GATE	

LPC Bus Topology

The LPC interface uses PCI electrical specifications. Overall bus length is limited by the flight time allocated for a transfer in any given clock cycle.

The time allocated for signal flight time is approximately 10 ns when PCI clock skew is taken into account. PCI driver strength does not allow first incidence switching, which means it can take up to 3 bus flight times for a signal to transition to a valid electrical level.

The proposed topology for the LPC bus using the debug port cable connector is specified in Table 3 below. The proposed topology will need to be validated using best know simulation and lab measurement techniques.

Table 3: LPC Bus Topology

Bus component	Max Length (Manhattan)	Max Flight Time	Total Budget	Note
Motherboard	6 inches	0.25 ns / inch	4.5 ns	Only loads are Flash , Debug connector, and chip set.
Cable	8 inches	0.20 ns /inch	4.8 ns	Includes length of connector
Module	0.5 inches	0.22 ns/inch	0.35 ns	Assumes only one load on module.
Total:			9.63 ns	

Cable Requirements

The cable can be built with standard 28 AWG, 50 mil pitch, flat cable. Reference cable connectors and cable part numbers are specified in Table 4.

Table 4: Cable Specification

Cable Type	Manufacturer	IDC Connector	Strain Relief	Cable
Legacy	FOXCONN	SE04107-04	001-1004-149	014-1020-002
Debug Only	FOXCONN	SE04087-04		014-1016-002

BIOS Requirements

BIOS is required to perform the following functions to support the debug module:

- Module Detection
- Debug Port initialization
- Debug Port ACPI Table reporting
- Optional legacy keyboard/mouse enabling
- Support ACPI Sleep State Transitions

One possible BIOS flow chart is shown in Figure 7 below. The code flow in the BIOS POST and BIOS BOOT process blocks will be different depending on the ACPI state transition occurring in the system.

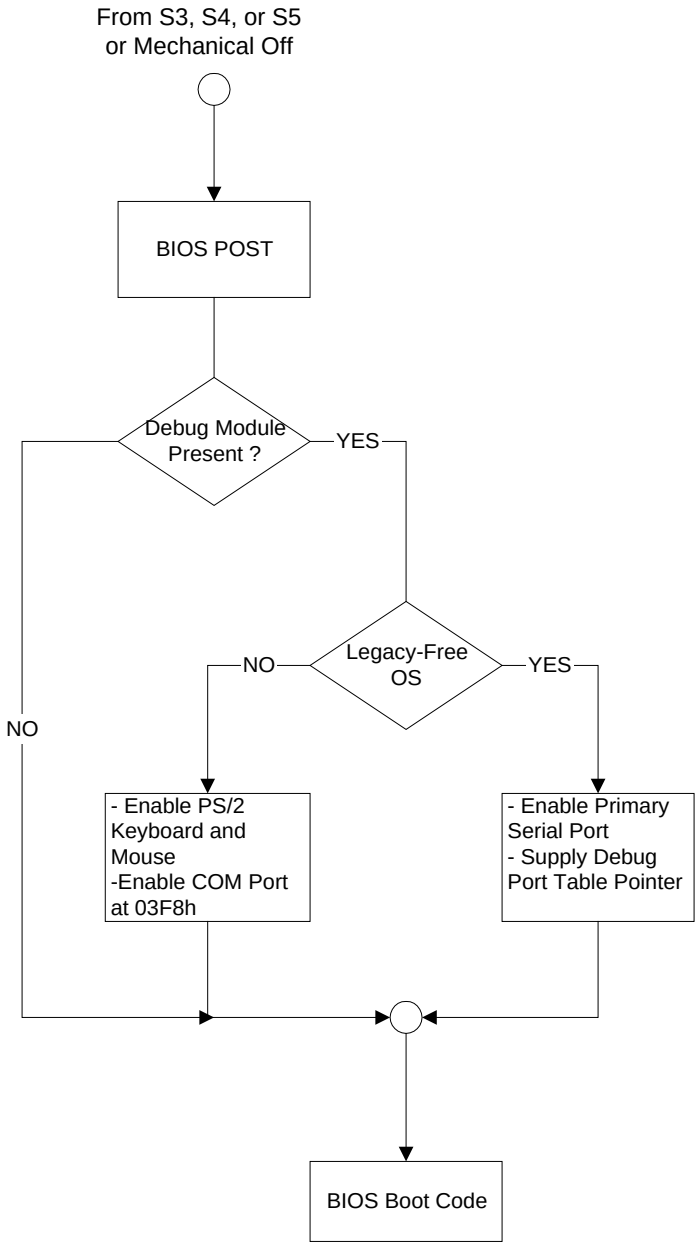

Figure 7: BIOS Flow

Module Detection

Using SMBus protocol, the BIOS must search for a serial EEPROM at addresses 1010111xb through 1010100xb. The EEPROM will contain a unique 16 bit ID at the start of the data structure. The Data structure is defined later in this document.

The EEPROM uses the following unique 16 bit ID at Word offset Zero in the EEPROM: **3C96H.**

The EEPROM contains the parameters required to enable and disable configuration of the SIO PnP registers.

Debug Port Initialization

The serial communication port must have a Plug and Play compatible register set in order to meet the BIOS and OS programming requirements of this specification. The following information will be provided to the BIOS by the serial EEPROM on the module:

- Primary Serial Port Logical Device Number
- Configuration Port address and entry sequence
- Primary Serial Port Mode register value that meets this specification
- Configuration Exit sequence

Debug Port Reporting

Once the primary serial port has been enabled and programmed with a valid I/O base address, the BIOS will enable the ACPI Debug Table. For more information on programming the Table see:

http://www.microsoft.com/hwdev/onnow/download/LFreeACPI.doc

The BIOS is not responsible for programming the 16550 compatible UART registers. This will be done by operating system software.

Legacy Enable

If the platform supports a debug module with the legacy extension, the BIOS requires a legacy mode option in its setup. The EEPROM will contain the necessary information to allow the BIOS to enable legacy keyboard and mouse operation. No other legacy peripheral interfaces are supported.

When the setup option for legacy mode is enabled, the BIOS must set the base address of the primary serial port to x03f8h.

Support ACPI Sleep State Transitions

The BIOS is responsible for reprogramming the SIO on the module following S3, S4 and S5 exit transitions. The debug module is not on standby power and register contents will be lost during those sleep states.

EEPROM Data Structure

This section contains information on how to program the EEPROM device on the Debug Module. The EEPROM allows interoperability between modules with different SIO devices and baseboards that have a common software driver interface.

EEPROM Organization

The EEPROM must be a byte-organized device to meet the requirements of the Industry standard SMBus protocol, which requires byte access capability. The Debug Module data structure is organized as Words (16 bits quantities).

The overall Map of the EEPROM is given in Table 5 below.

Table 5: EEPROM MAP

Size Bytes	Name	Description
32	Header Record	Contains offsets to other Records
16	SIO I/O Access Record	Port Addresses and access/exit parameters
16	Alternate SIO I/O Access Record	Port Addresses and access/exit parameters
16	Primary Serial Port Record	PnP programming parameters
16	Keyboard Record	PnP programming parameters
16	SIO Device Record	ID and Revision information
16	Module Physical Record	ID and Revision information

The Debug Module data structure requires a fixed size of 128 bytes. If an EEPROM larger than 128 bytes is used, then the extra space is unclaimed. Typically a 1K bit or 2K bit EEPROM will be used in this application.

A record is always allocated its size, in bytes, whether or not it is implemented. Records that are not implemented must have an offset of Zero in their header record descriptor.

The BIOS must always use the offset provided in the header for a particular record. The BIOS must not assume a record is always at a fixed location in the EEPROM. This restriction is necessary to allow revisions to the Debug Module data structure.

Header Record

This record provides information about the EEPROM device and provides physical pointers (offsets) to the other record types in the EEPROM. EEPROM space is allocated for all records in the header. If a record has an offset value of 0000, then the data in that record is not valid and should not be used.

Table 6: Header Record

Word Address (Hex)	Byte Offset		Values (Hex)	Notes
	1	**0**		1
0	EEPROM ID, MSB	EEPROM ID, LSB	3C96	2
2	EPROM bytes Allocated, MSB	EPROM bytes Allocated, LSB	0100	3
4	EPROM Size, bytes, MSB	EPROM Size, bytes, LSB	0000-FFFF	4
6	Reserved	Configuration Port record Descriptor	0020	5
8	Reserved	Alternate Configuration Port record Descriptor	0000, 0030	6
A	Reserved	Primary Serial Port Record Descriptor	0040	7
C	Reserved	8042 Record Descriptor	0000, 0050	8
E	Reserved	SIO Record Descriptor	0060	9
10	Reserved	Module Physical Record Descriptor	0070	10
12-1C	Reserved	Reserved	0000	11
1E	Zero Checksum MSB	Zero Checksum LSB	0000-FFFF	12

1. EEPROM Byte address = Word Address + byte offset.
2. This is the ID that identifies the Debug Module to the BIOS. System Board designers must make sure that this ID does not conflict with other EEPROM devices on the I2C bus.
3. This is the number of bytes in the debug Module Data structure, which is 128 bytes in length.
4. This is the number of physical bytes in the EEPROM device. Unused storage, in bytes, is the EEPROM Size minus Bytes Allocated.
5. This is the offset, from location 0h, to the configuration access port record for the SIO.
6. This is the offset, from location 0h, to the alternate configuration access port record for the SIO. This record is provided in case the primary location conflicts with another motherboard device port address. Not all debug modules will implement an alternate record. A value of 0000h means this record is not valid.
7. This is the offset, from location 0h, to the record for the primary serial port. It contains the PnP LDN for the primary serial port and programming parameters. This record is required.
8. This is the offset, from location 0h, to the record for the 8042 Keyboard controller, if present. It contains the PnP LDN for the keyboard controller and programming parameters. A value of 0000h means there is no keyboard controller present on the module.
9. This is the offset, from location 0h, to the SIO descriptor record. This Record is required. It supplies a chip ID and a revision number for the SIO on the debug module.
10. This is the offset, from location 0h, to the Module Physical descriptor record. This Record is required. It supplies debug module manufacturing information to the BIOS.
11. These locations are reserved. Reserved locations always contain Zeros.
12. This is a 16 bit value that results in a Zero checksum over the Header Record. The zero checksum is calculated by performing an unsigned addition, modulo 2**16 (65,536), over EEPROM Word locations 0000 – 0x1C, taking the 2's complement of that number, and then adding one (1).

Configuration Access Port Record

The Configuration Access Port Record Format is the same for the primary access port and the alternate access port. Not all SIO devices support an alternate access port. The BIOS will use the alternate access port if there is a conflict for the primary access port address. If an address conflict with another motherboard device cannot be resolved, the conflicting debug module cannot be used in that system (See Error Reporting).

Table 7: Configuration Access Port Record

Word Address (Hex)	**Byte Offset (Hex)**		**Value (Hex)**	**Notes**
	1	**0**		1
0	Index Port I/O Address, MSB	Index Port I/O Address, LSB	0000-0100	2
2	Data Port I/O Address, MSB	Data Port I/O Address, LSB	0000-FFFF	2
4	Reserved	Access Key Length, Bytes	0000-0004	3
6	Key1	Key 2	0000-FFFF	4
8	Key3	Key4	0000-FFFF	4
A	Exit Key, Index Port	Exit Key, Data Port	0000-00FF	5
C	Reserved	Reserved	0000	6
E	Reserved	Reserved	0000	

1. Byte address = Word Address + byte offset
2. This Address should be in the I/O range (x000 –x3FFh) and should not conflict with chip set fixed I/O ranges, such as for DMA registers, PIC and PIT.
3. The Access Key is a set of byte values that must be written to the Index Port address to enable the SIO for programming. A value of Zero means no key is required and that the SIO is accessible. The Maximum allowed key length is 4 bytes.
4. The keys must be written sequentially to the Index port, beginning with Key1 and ending with KeyN, where N = Access Key length.
5. The Index Port Exit Key is a byte value that must be written to the Index Port address to disable the SIO for programming. A value of Zero means no key is required and that the SIO cannot be disabled. In that case, the BIOS cannot re-use the Index/Data port addresses and should reserve these addresses using PnP ID PNP0C02.
6. The Data Port Exit Key is a byte value that must be written to the Data Port address to disable the SIO for programming. This value may be a “don’t care” for some SIO devices. A value of Zero should be supplied if this action is not required.

Primary Serial Port Record

This record is used to program the serial port. The SIO must contain a standard 16550 SERIAL PORT. The data in this record provides the parameters necessary to:

- Locate the serial port plug and play register space (LDN)
- Enable / disable the serial port
- Program the serial Port base register location
- Program the Interrupt line for the serial port
- Program any SIO-specific mode registers for the serial port

Table 8: Primary Serial Port Record

Word Address (Hex)	Byte Offset (Hex)		Value (Hex)	Notes
	1	**0**		1
0	Reserved	LDN	0000-00FF	2
2	SERIAL PORT Disable Value, index 30h	SERIAL PORT Enable Value, index 30h	0000-FFFF	3
4	Index 60 base address value I/O Base LSB	Index 61 base address value I/O base MSB	F803	4
6	Reserved	Reserved	0000	4
8	Interrupt Control Value, index 70h	Interrupt Control Value, index 71h	0000-00FF	5
A	Interrupt Control Value, index 72h	Interrupt Control Value, index 73h	0000-00FF	5
C	Device specific value, index F1h	Device specific value, index F0h	0000-00FF	6
E	Device specific value, index F3h	Device specific value, index F2h	0000-00FF	6

1. Byte address = Word Address + byte offset
2. This is the PnP register Logical Device Number. The LDN is programmed into the global Logical Device number register at global configuration index 07h to access that device. The primary serial port LDN will be different for different SIO devices.
3. These byte values are written to the Activation control register at register index 30h of the PnP device. The Enable value is used to enable the SERIAL PORT. The disable value is used to disable the SERIAL PORT.
4. PnP Index registers 60,61h contain the base address registers for the SERIAL PORT. The EEPROM contains the address that should be used for legacy mode operation. It is the responsibility of the BIOS to assign an acceptable, legacy-free, debug port base address to the Primary COM port in the SIO.
5. The Interrupt parameters supplied in offset 8h through Ah for index registers 70h through 73h, must result in no interrupt being routed to the motherboard from the module.
6. These byte values defined for PnP logical device index registers F0h through F3h and are SIO device specific. The values supplied must result in proper 16550 SERIAL PORT operation. An example of a parameter that may need to be specified is the master baud clock rate.

Keyboard Controller (8042) Record

This is an optional record. It will only be provided by modules that contain a keyboard controller. The keyboard controller must have a device activation register. The keyboard controller must be disabled by the BIOS when the module is being used with a legacy free operating system. This means the BIOS must have a setup option for enabling and disabling this feature.

Table 9: Keyboard Controller Record

Word Address (Hex)	**Byte Offset (Hex)**		**Value (Hex)**	**Notes**
	1	**0**		1
0	Reserved	LDN	0000-00FF	2
2	Keyboard Disable Value, index 30h	Keyboard Enable Value, index 30h	0000-FFFF	3
4	Index 61 base address value (I/O base, LSB)	Index 60 base address value (I/O base MSB)		4
6	Index 63 base address value (I/O base LSB)	Index 62 base address value(I/O base MSB)		4
8	Interrupt Control Value, index 71h	Interrupt Control Value, index 70h	0000-00FF	5
A	Interrupt Control Value, index 73h	Interrupt Control Value, index 72h	0000-00FF	5
C	Device specific value, index F1h	Device specific value, index F0h	0000-00FF	6
E	Device specific value, index F3h	Device specific value, index F2h	0000-00FF	6

1. Byte address = Word Address + byte offset
2. This is the PnP register Logical Device Number. The LDN is programmed into the global Logical Device number register at global configuration index 07h to access that device.
3. These byte values are written to the Activation control register at register index 30h of the PnP device. The Enable value is used to enable the keyboard and mouse functions. The disable value is used to disable the keyboard and mouse functions.
4. The byte values supplied by the EEPROM for PnP device index registers 60h through 63h must result in the keyboard controller data register being mapped to I/O address 0060h and the control register being mapped to I/O address 64h.
5. The byte values for PnP device index registers 70h through 73h are used to define the interrupt mapping for the keyboard and mouse. The values provided by the EEPROM should result in standard mouse / keyboard interrupt routing as specified in the PC99 System Design Guide.
6. These byte values defined for PnP device index registers F0h through F3h and are SIO device specific. The keyboard reset signal and the A20 GATE signal must be programmed for 8042 as the source and Port 92 access should be disabled in the SIO.

SIO Device Record

This record is used to identify the manufacturer, device number, and device revision number for the SIO. This data can be used to control compatibility problems between the motherboard BIOS and debug modules.

Table 10: SIO Device Record

Word Address (Hex)	**Byte Offset (Hex)**		**Value (Hex)**	**Notes**
	1	**0**		1
0	C1	C0: Right most character	2020	2
2	C3	C2	2020	2
4	C5	C4	4520	2
6	C7: Left Most character	C6	4954	2
8	Device ID LSB	Device ID MSB	0000-FFFF	3
A	Reserved	Reserved	0000	
C	Reserved	Device Revision	0000-00FF	4
E	Reserved	Reserved	0000	

1. Byte address = Word Address + byte offset
2. SIO Manufacturers name in ASCII. The name is read, left to right. Allowed characters are A-Z, caps only, blank, and underscore. There are 8 characters Maximum. Trailing blanks must be used in names of less than 8 characters. The Example value provided is: C7::C0 = ITEbbbbb, where b=blank position.
3. This is the offset into the global plug and play register set that contains the MSB of the DEVICE ID.
4. This is the offset into the global plug and play register set that contains the device revision. This value must be non-zero.

Module Physical Record

This record is used to identify the manufacturer, model number, and revision number for the module and EEPROM. This data can be used to control compatibility problems between the motherboard BIOS and debug modules.

Table 11: Module Physical Record

Word Address (Hex)	**Byte Offset (Hex)**		**Value (Hex)**	**Notes**
	1	**0**		1
0	C1	C0: Right most character	2020	2
2	C3	C2	4C20	2
4	C5	C4	5445	2
6	C7: Left Most character	C6	494E	2
8	Model Number, D3^D2	Model Number D1^D0	0000-FFFF	3
A	Model Number, D7^D6	Model Number D5^D4	0000-FFFF	3
C	Reserved	Module Revision	0000	4
E	Reserved	EEPROM Revision	0000	4

1. Byte address = Word Address + byte offset
2. Manufacturers name in ASCII. The name is read, left to right. Allowed characters are A-Z, caps only, blank, and underscore. There are 8 characters Maximum. Trailing blanks must be used in names of less than 8 characters. The Example value provided is: C7::C0 = INTELbbb, where b=blank position.
3. Model Number is a packed, 8 digit, BCD number, with D7 being the left most digit and D0 being the right most digit.
4. This is a 16 bit, binary value.

Sample EEPROM Programming for the ITE8761E

The following Table specifies the EEPROM programming for the Integrated Technology Express, Inc . IT8761E LPC Super I/O device. The EEPROM device in this example is a 2K bit, 256 x 8 device. The debug module described is one that supports the legacy extension functions.

Table 12: IT8761E EEPROM

Word Address (Hex)	Byte Offset (Hex)		Value (Hex)	Notes
	1	**0**		
0	EEPROM ID, MSB	EEPROM ID, LSB	3C96	
2	EPROM bytes used, MSB	EPROM bytes used, LSB	0100	
4	EPROM Size, bytes, MSB	EPROM Size, bytes, LSB	0200	
6	Configuration Port record Offset MSB	Configuration Port record Offset LSB	0020	
8	Alternate Configuration Port record Offset MSB	Alternate Configuration Port record Offset LSB	0030	
A	Primary Serial Port record offset , MSB	Primary Serial Port LDN offset , LSB	0040	
C	8042 record offset, MSB	8042 Record Offset, LSB	0050	
E	SIO record offset, MSB	SIO record offset, LSB	0060	
10	Module Physical Record offset, MSB	Module Physical Record offset, LSB	0070	
12-1C	Reserved	Reserved	0000	
1E	Zero Checksum MSB	Zero Checksum LSB	BDBA	
20	Index Port I/O Address, MSB	Index Port I/O Address, LSB	002E	
22	Data Port I/O Address, MSB	Data Port I/O Address, LSB	002F	
24	Reserved	Access Key Length, Bytes	0004	
26	Key1	Key2	8761	
28	Key3	Key4	5555	
2A	Reserved	Exit Key Index Port	0002	
2C	Key2	Exit Key, Data Port	0002	
2E	Reserved	Reserved	0000	
30	Index Port I/O Address, MSB	Index Port I/O Address, LSB	004E	
33	Data Port I/O Address, MSB	Data Port I/O Address, LSB	004F	
34	Reserved	Access Key Length, Bytes	0004	
36	Key1	Key2	8761	
38	Key3	Key4	55AA	
3A	Exit Key, Index Port	Exit Key, Data Port	0002	
3C	Reserved	Reserved	0002	
3E	Reserved	Reserved	0000	
40	Reserved	LDN	0001	
42	SERIAL PORT Disable Value, index 30h	SERIAL PORT Enable Value, index 30h	0001	
44	Index 61 base address value	Index 60 base address value	F803	

Word Address (Hex)	Byte Offset (Hex)		Value (Hex)	Notes
	1	0		
46	Reserved	Reserved	0000	
48	Interrupt Control Value, index 71h	Interrupt Control Value, index 70h	0000	
4A	Interrupt Control Value, index 73h	Interrupt Control Value, index 72h	0000	
4C	Device specific value, index F1h	Device specific value, index F0h	0000	
4E	Device specific value, index F3h	Device specific value, index F2h	0000	
50	Reserved	LDN	0000	
52	Keyboard Disable Value, index 30h	Keyboard Enable Value, index 30h	0003	
54	Index 61 base address value	Index 60 base address value	6000	
56	Index 63 base address value	Index 62 base address value	6400	
58	Interrupt Control Value, index 71h	Interrupt Control Value, index 70h	0201	
5A	Interrupt Control Value, index 73h	Interrupt Control Value, index 72h	0000	
5C	Device specific value, index F1h	Device specific value, index F0h	0000	
5E	Device specific value, index F3h	Device specific value, index F2h	0000	
60	C1	C0: Right most character	2020	
62	C3	C2	2020	
64	C5	C4	4520	"ITE"
66	C7: Left Most character	C6	4954	
68	Device ID LSB	Device ID MSB	2120	
6A	Reserved	Reserved	0000	
6C	Reserved	Device Revision	0022	
6E	Reserved	Reserved	0000	
70	C1	C0: Right most character	2020	OPSD
72	C3	C2	2020	
74	C5	C4	5344	
76	C7: Left Most character	C6	4F50	
78	Model Number, D3^D2	Model Number D1^D0	9079	
7A	Model Number, D7^D6	Model Number D5^D4	00A0	
7C	Reserved	Module Revision	0001	
7E	Reserved	EEPROM Revision	0000	

Error Conditions

The errors in Table 13 should be reported to the user for failure analysis. The reporting mechanism is system dependent.

Table 13: Error Condition List

Item	Description	System Response
1	Module could not be allocated a unique configuration Port address.	Report Error through POST Code mechanism.
2	Module EEPROM was blank (contained all Ones or All Zeros)	Report Error through POST Code mechanism.
3	SIO could not be configured. Access port/key failure	Report Error through POST Code mechanism.
4	Checksum failure	Report Error through POST Code mechanism.

RS-232C Requirements

The debug module connects the PC under test to a remote debug console. Because this connection is a computer to computer connection (no intervening modems), a null modem cable is required to cross-connect the data lines and to cross-connect certain control signals.

16550 Compatible Register Set

The Serial Port as exposed to the OS through the I/O port locations specified in the I/O base registers must have a 16550 register interface. The location and type of UART is specified in the DEBUG Port Table.

RS-232 Required Signals

The Module uses a standard DB-9 Male Connector for its RS-232C Interface connection. The Connector Pin out is specified as follows:

Table 14: Module RS-232C Connections

Pin Number	Signal Name	Type	Module Connection	Description
1	DCD	Input	YES	Carrier Detect
2	RXD	Input	YES	Receive Data
3	TXD	Output	YES	Transmit Data
4	DTR	Output	YES	Data Terminal Ready
5	SG	-	YES	Signal Ground
6	DSR	Input	YES	Data Set Ready
7	RTS	Output	YES	Request to Send
8	CTS	Input	YES	Clear To Send
9	RI	Input	YES	Ring Indicator
Shell	Chassis Ground	-	YES	Chassis Ground

The module requires 3 RS-232C drivers and 5 RS-232C receivers.

RS232-C Null Modem Cable

The following standard, null modem connections are required. This cable type is available from many serial cable vendors.

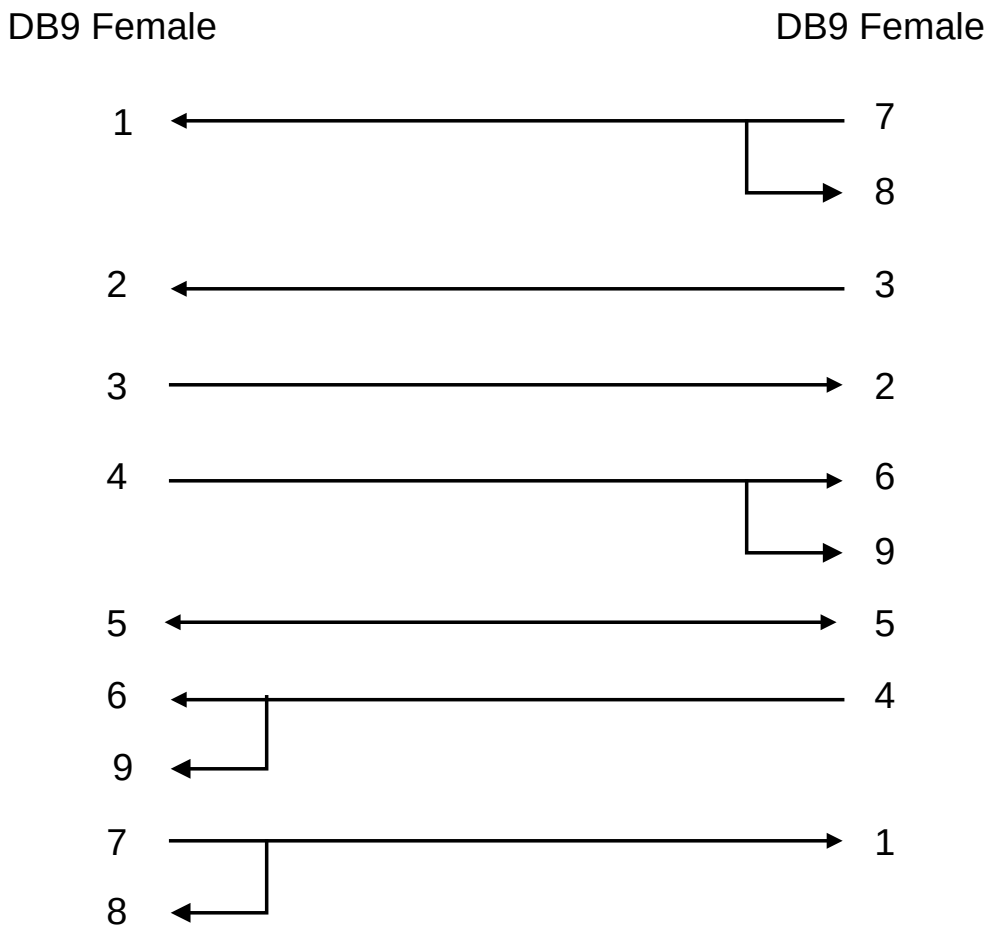

Figure 8: Null Modem Cable Pin Connections